PATH FOLLOWER MOBILE ROBOT USING PID CONTROLLER

MUHAMMAD BIN MAZLAN

This thesis is submitted as partial fulfillment of the requirements for the award of the Bachelor of Electrical and Electronic Engineering (Hons.) (Electronics)

Faculty of Electrical & Electronics Engineering
Universiti Malaysia Pahang

NOVEMBER, 2008

ACKNOWLEDGEMENT

First and foremost, I am very grateful to the almighty ALLAH S.W.T for giving me the key and opportunity to accomplish my Final Year Project.

This work would not have been possible without the support and encouragement of my supervisor, En Ahmad Nor Kasruddin, under whose supervision I chose this topic and began the thesis, my advisor in the final stages of the work, has also been abundantly helpful, and has assisted me in numerous ways, including guiding to completing my final year project.

I cannot end without thanking my family, on whose constant encouragement and love I have relied throughout my time at the university. I am grateful also to the examples of my father and my mother. Their unflinching courage and conviction will always inspire me, and I hope to continue, in my own small way, the noble mission to which they gave their lives. It is to them that I dedicate this work.

ABSTRACT

This project is about path follower mobile robot using PID Controller. As we know the PID controller is a generic control loop feedback mechanism widely used in industrial control system. The controller corrects the error that makes the mobile robot moving out of track. This project concentrates in the development path follower mobile robot which is moving in square path with straight line and turn 90 degree and integrating the PID Controller into steering path for the path follower mobile robot to make the mobile robot moving smooth straight line and turning 90degree. At the end of this project also discuss about the comparison between controllers that could integrate into the mobile robot system.

ABSTRAK

Projek ini menerangkan berkenaan robot mudah alih pengikut jalan menggunakan pengawal PID. Seperti sedia maklum, pengawal PID adalah mekanisme pengawal gelung suapbalik generik yang banyak digunakan dalam industri. Pengawal ini bertindak membetulkan kesalahan yang boleh menyebabkan robot mudah alih ini terkeluar dari landasan. Projek ini memfokuskan pada pembangunan robot mudah alih pengikut jalan di mana ia akan bergerak dalam jalan segi empat sama dengan jalan lurus dan 90 darjah belok, dan menyatukan pengawal PID ke dalam steering robot mudah alih pengikut jalan untuk menjadikan robot mudah alih dapat bergerak lancar lurus dan membelok 90 darjah. Di pengakhiran projek ini juga membincangkan tentang perbandingan antara pengawal-pengawal yang boleh di satukan ke dalam system robot mudah alih.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
1.	**INTRODUCTION**	1
	1.1 Overview	1
	1.2 Objective	2
	1.3 Scope Of Project	3
	1.4 Problem Statement	3
	1.5 Thesis Organization	4
2.	**LITERATURE REVIEW**	5
	2.1 Project Review	5
	2.2 Proportional-Integral-Derivative (PID)	6
	2.3 Mobile Robot	8
	2.4 PIC Microcontroller	8
	2.5 DC Motor	9
	2.6 Navigation System	11
	2.7 Computer Programming	12
3.	**METHODOLOGY**	14
	3.1 Hardware configuration	14
	3.2 Software Configuration	20

4.	**RESULT AND DISCUSSION**	27
	4.1 Result	27
	4.2 Path Follower Mobile Robot	28
	4.3 PIC Microcontroller Basic Circuit	29
	4.4 Infra Red Sensor Circuit	30
	4.5 Direct Current Motor Circuit	31
	4.6 Discussion	32
	4.7 Costing and Commercialization	33
5.	**CONCLUSION AND FUTURE RECOMMENDATION**	34
	5.1 Conclusion	34
	5.2 Future Recommendation	35
REFERENCE		36
Appendices A-D		37-53

LIST OF FIGURE

FIGURE NO.	TITLE	PAGE
2.1	A block diagram of PID controller	7
2.2	PIC 16F877A pin configuration	9
2.3	DC motor 12V	11
2.4	The Output System Connection	11
2.5	Hierarchy Of Computer Programming	13
3.1	Power Supply Modules	15
3.2	Example of crystal to generate clock	15
3.3	Basic Circuit PIC	16
3.4	Pin configuration L293B	17
3.5	The Infrared Red Sensor Using LM324	18
3.6	Whole Module Circuit	19
3.7	MPLAB Software	22
3.8	MPASM Software	23
3.9	PICKit 2 v2.4 software	24

4.1	The overall circuit with prototype of the mobile robot.	25
4.2	Basic PIC Microcontroller Circuit.	29
4.3	IR Sensor Circuit	30
4.4	DC Motor Circuit	31
4.5	The complete one board circuit	32

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
A	Datasheet of PIC16F877A1	38
B	Datasheet of L293B	39
C	Datasheet of LM324	40
D1	Example of Program by using Assembly	41
D2	Example of List File	45
D3	Example of Error File	53

CHAPTER 1

INTRODUCTION

1.1 Overview

Nowadays, there are many mobile robots in different shape, size and application. Mobile robot with programmed path for movement is one of them. Many programmed path robot just hardware and program. A mobile robot is an automatic machine that is capable of movement in a given environment. Mobile robots have the capability to move around in their environment and are not fixed to one physical location. In contrast, industrial robots usually consist of a jointed arm (multi-linked manipulator) and gripper assembly (or end effectors) that are attached to a fixed surface. Mobile robots are the focus of a great deal of current research and almost every major university has one or more labs that focus on mobile robot research. Mobile robots are also found in industry,

military and security environments. They also appear as consumer products, for entertainment or to perform certain tasks like vacuum cleaning or mowing.

The one way mobile robot which is when we turn on the mobile robot, the mobile robot will move and do the task we ask. In control term is known as open loop system. Therefore, I invented this mobile robot with programmed path with PID controller to upgrade the open loop system in robot into closed loop system mobile robot. The implementation of PID is to improve the controller circuit based on the data from the experiment.

There are many controllers in the industry such as PID, Fuzzy Logic, Artificial Neural Network and Linear Quadratic Regulator.

1.2 OBJECTIVE

At the end of this project:

i. Able to develop path follower mobile robot.
ii. Able to design PID in microcontroller to steering path follower mobile robot.
iii. Comparing the result the path follower mobile robot using PID and without PID.

1.3 SCOPE OF PROJECT

Scopes the need to be proposed are;

i. Making path follower mobile robot.
   - Making normal mobile robot with square path.

ii. Implement PID in path follower mobile robot.
   - After accomplish the mobile robot, run a experiment and correct its error by using PID controller

iii. Making comparison between PID system and others.
   - Finalized and making comparison between path follower mobile robot without PID and with PID.

1.4 PROBLEM STATEMENT

The main problem in making path follower mobile robot is easily move out of track when turning at the corner. Some robot cannot move straight even we static the speed of the robot. Most of the problem we can repair through program and hardware but the system is not complete and have condition to make the system is stabilized. Therefore, we need to design a system to stabilize the basic system without and condition to make is stable and here came PID controller.

1.5 THESIS ORGANIZATION

There are 5 chapters in this thesis including this chapter. The content of each chapter which are:

i. In chapter 2, contain detailed description each part of my project. It will be discussion about, overall project overview, Proportional-Integral-Derivative and path follower mobile robot.
ii. Next in chapter 3, discuss about project methodology. This methodology I used in the project.
iii. Then in chapter 4 is result and discussion. It will tell result about this project and costing & commercialization
iv. Lastly in chapter 5 is conclusion and future recommendation.

CHAPTER 2

LITERATURE REVIEW

2.1 PROJECT REVIEW

This project will be design the path follower mobile robot with PID as the controller. A system for path tracking of mobile robot is a program in the PIC is forward 20cm and turn 90degree for 5cm radius and the sensor at the front edge mobile robot. [3] This project is also making turn at maximum speed without move out from the track. Traditional method is making turn using remote control or program with fixed speed. The multi loop nature of the architecture ensures adequate stability at different levels yielding safe navigation and accomplishment of higher level tasks.[1] Navigation is a major issue when addressing mobile robotics because the concept is so wide that it includes all aspects of directing a robot's course as it traverses the

environment.[1] The sensor for the development of the mobile robot path following algorithm is important.[2]

I have formulated the problem is to (1) create the system or controller that can follow desired or set point with variable speed, (2) keep the robot in the track even after one complete square, (3) the repeated program that I will be used to keep mobile robot on track which is move forward and turn 90degree and response to the error. The approach that I have taken is to develop normal mobile robot with square path movement and I add infra red sensor at the front edge of the robot. This sensor will detect the line of the track and send error signal to the controller. To maintain the mobile robot on the course is using rubber tire and light weight body material. The program will have the main program and the interrupt program for correct the movement of the mobile robot which is respond to the error.

2.2 Proportional Integral and Derivative Controller (PID)

A proportional-integral-derivative controller (PID controller) is a generic control loop feedback mechanism widely used in industrial control systems. A PID controller attempts to correct the error between a measured process variable and a desired set point by calculating and then outputting a corrective action that can adjust the process accordingly.

The PID controller calculation (algorithm) involves three separate parameters; the Proportional, the Integral and Derivative values. The Proportional value determines the

reaction to the current error, the Integral determines the reaction based on the sum of recent errors and the Derivative determines the reaction to the rate at which the error has been changing. The weighted sum of these three actions is used to adjust the process via a control element such as the position of a control valve or the power supply of a heating element.

By "tuning" the three constants in the PID controller algorithm the PID can provide control action designed for specific process requirements. The response of the controller can be described in terms of the responsiveness of the controller to an error, the degree to which the controller overshoots the set point and the degree of system oscillation. Note that the use of the PID algorithm for control does not guarantee optimal control of the system.

Some applications may require using only one or two modes to provide the appropriate system control. This is achieved by setting the gain of undesired control outputs to zero. A PID controller will be called a PI, PD, P or I controller in the absence of the respective control actions. PI controllers are particularly common, since derivative action is very sensitive to measurement noise, and the absence of an integral value prevents the system from reaching its target value due to the control action.

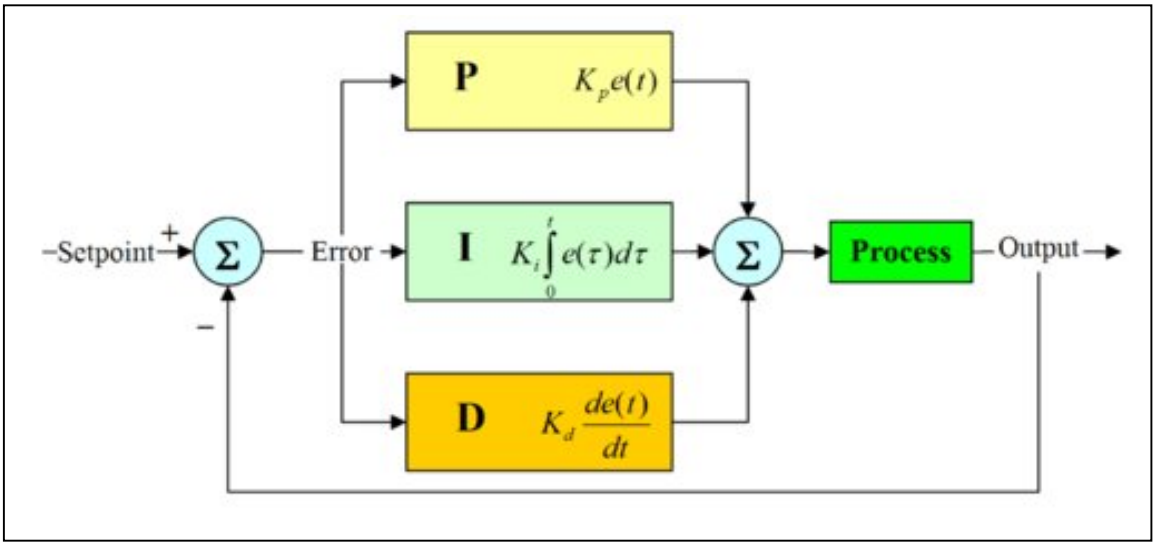

Figure 2.1: A block diagram of PID controller

2.3 Mobile Robot

A Mobile Robot is an automatic machine that is capable of movement in a given environment. Mobile robots have the capability to move around in their environment and are not fixed to one physical location. In contrast, industrial robots usually consist of a jointed arm (multi-linked manipulator) and gripper assembly (or end effectors) that is attached to a fixed surface. Mobile robots are the focus of a great deal of current research and almost every major university has one or more labs that focus on mobile robot research. Mobile robots are also found in industry, military and security environments. They also appear as consumer products, for entertainment or to perform certain tasks like vacuum.

2.4 PIC Microcontroller

The name of PIC initially referred to “Programmable Interface Controller” [4], but shortly thereafter was renamed “Programmable Intelligent Computer” [5].

PICs are popular with developers and hobbyists alike due to their low cost, wide availability, large user base, extensive collection of application notes, availability of low cost or free development tools, and serial programming (and re-programming with flash

memory) capability. Microchip recently announced the shipment of its six billionth PIC processor.

Microcontroller that I have use is PIC 16F877 which have 3Timer; Timer0 and Timer 2 for 8 bit counter or timer, Timer1 for 16 bit counter or timer, 2 Capture, Compare, PWM modules, 10bit multi-channel Analog-to-Digital converter, Synchronous Serial Port (SSP), Universal Synchronous Asynchronous Receiver Transmitter (USART/SCI) with 9 bit address detection, Parallel Slave Port (PSP) 8 bit, and 368bytes memory with 256 bytes EEPROM.

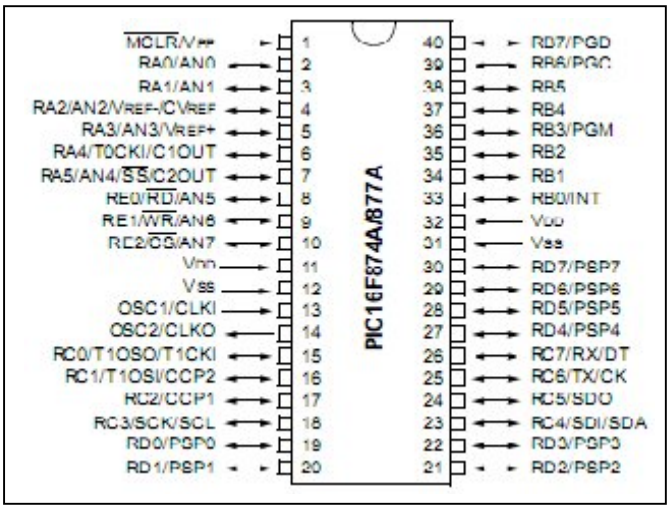

Fig 2.2: PIC 16F877A pin configuration

2.5 DC motor

A DC motor works by converting electric power into mechanical work. This is accomplished by forcing current through a coil and producing a magnetic field that spins

the motor. The simplest DC motor is a single coil apparatus, used here to discuss the DC motor theory.

The voltage source forces voltage through the coil via sliding contacts or brushes that are connected to the DC source. These brushes are found on the end of the coil wires and make a temporary electrical connection with the voltage source. In this motor, the brushes will make a connection every 180 degrees and current will then flow through the coil wires. At 0 degrees, the brushes are in contact with the voltage source and current is flowing. The current that flows through wire segment C-D interacts with the magnetic field that is present and the result is an upward force on the segment. The current that flows through segment A-B has the same interaction, but the force is in the downward direction. Both forces are of equal magnitude, but in opposing directions since the direction of current flow in the segments is reversed with respect to the magnetic field. At 180 degrees, the same phenomenon occurs, but segment A-B is forced up and C-D is forced down. At 90 and 270-degrees, the brushes are not in contact with the voltage source and no force is produced. In these two positions, the rotational kinetic energy of the motor keeps it spinning until the brushes regain contact.

One drawback to the motor is the large amount of torque ripple that it has. The reason for this excessive ripple is because of the fact that the coil has a force pushing on it only at the 90 and 270 degree positions. The rest of the time the coil spins on its own and the torque drops to zero. The torque curve produced by this single coil, as more coils are added to the motor, the torque curve is smoothed out.

The resulting torque curve never reaches the zero point and the average torque for the motor is greatly increased. As more and more coils are added, the torque curve approaches a straight line and has very little torque ripple and the motor runs much more smoothly. Another method of increasing the torque and rotational speed of the motor is to increase the current supplied to the coils. This is accomplished by increasing the voltage that is sent to the motor, thus increasing the current at the same time.

Fig 2.3: DC motor 12V

2.6 Navigation System

Navigation for mobile robot is the main problem. Therefore, we need to calculate accurate and precisely to avoid the error regarding the path that we develop. We also use a sensor to detect the rotational of the tire. The path or course that we have made is square path. It have diameter $40cm^2$. Therefore, the mobile robot will move straight for 30cm and turn 90degree for circle with radius 2.5cm.

The overall system, which we use the controller is PID and we continue the process according to the set point without error. Figure 2.4 is shown the hardware of the project and the flow of the hardware works.

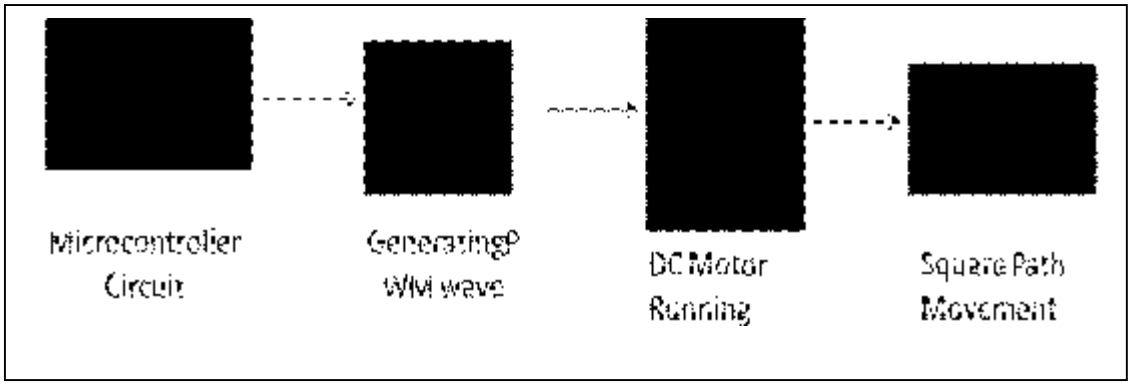

Fig 2.4: The Output System Connection.

2.7 Computer Programming

Programming or coding it is word familiar with the computer programming. Computer programming is the process of writing, testing, debugging/troubleshooting, and maintaining the source code of computer programs. The computer program is written in programming language which has many types in the world right now. Examples of programming language are Machine Code, Assembly, BASIC, C, JAVA, FOTRON, PHYTON and PERL. There are many more programming languages which are in the internet. [7] The programmer need to choose based on their expertise and experience.

Machine Code is based on binary number which is either '1' or '0'. This programming language is category in low level language which is the most possible programming language that can be program. This programming faster execution and can control every component in a system but complex programming and need to understand system configuration. The programmer needs to memorize every group of number to do an instruction. Differ Machine Code for each processor and normally is upward compatible. Machine Code only language the computer can understand.

Assembly Language is an instruction based on the processor. This programming required an assembler to assemble into machine code. This programming we can understand but the computer cannot. Assembly language is much easier to write the program than the Machine Code.

High Level languages are using instruction by the compiler. It's required to follow the syntax and semantics. The problem with incompatible processor is solved by using this programming language. The high level languages have many types such as BASIC, C, FOTRON, PERL and JAVA. This programming language closed to our normal language. Therefore we can easily understand the program.

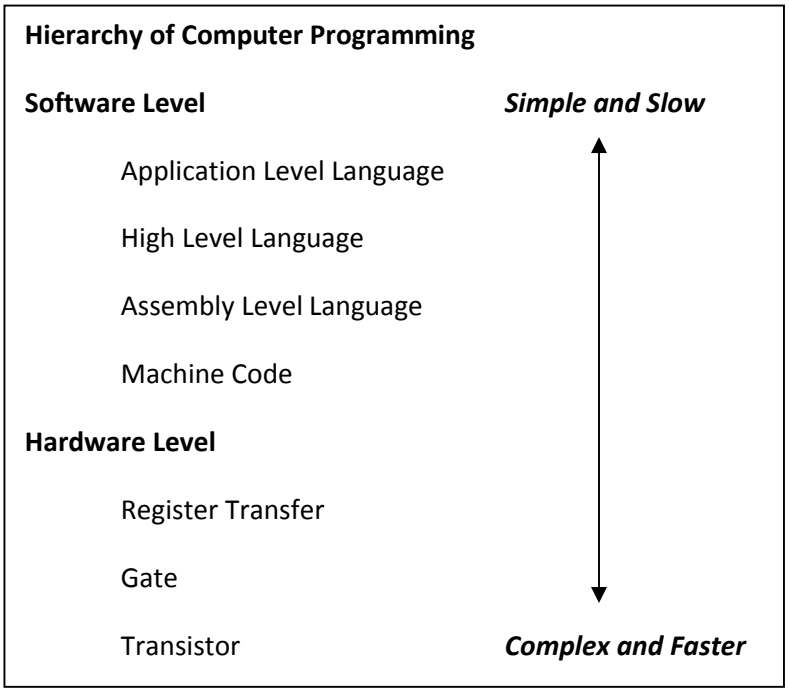

Figure 2.5: Hierarchy of Computer Programming

The computer programming is the process to make the program for computer to understand what the task and instruction that the computer need to do. The only computers understand is only Machine Code. Therefore the programming language other the Machine Code need an assembler (for Assembly Language) and the compiler (for the High Level Language) to convert the source code or program into Machine Code.

CHAPTER 3

METHODOLOGY

3.1 Hardware Configuration.

Hardware Installation

For hardware design, first is to design the power supply module which is to supply 5V fixed to PIC. Power supply module is importance to PIC to prevent damage if users give the higher input supply to device. The schematic diagram for power supply module is like in figure 3.1. Input to the power supply must greater than 7V to 7805 voltage regulator IC to achieve the 5V output supply to PIC and max232.

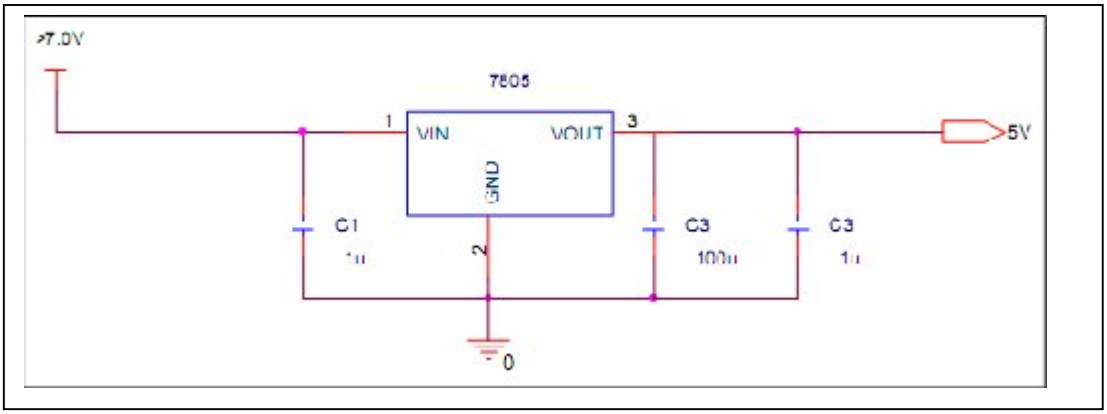

Figure 3.1: Power Supply Modules

The power module wasn't the only module to complete the basic circuit. There are another 2 circuit's module to complete the basic PIC circuit diagram, the clock circuit and the reset button circuit.

The clock circuit is to generate the clock pulse to PIC Microcontroller to operate. The range of the crystal is 4MHz until 20MHz. For low crystal clock value which is 4MHz we need 2 capacitors to smooth the pulse and stabilized the pulse. The range value of the capacitor is 10nF-33nF.

Figure 3.3: Example of crystal to generate clock.

The reset circuit is important even though it for system to revert to the initial state but it also for the PIC Microcontroller to start operate. The PIC Microcontroller will operate if only the Reset (MCLR pin 1) received or get the input high (4.5V - 5.0V) and the PIC will reset if only get the input low (below 4.5V). For high clock frequency,